Prepared for: Public Works and Government Services

Prepared by: Artificial Intelligence Management and Development Corporation
AIM (AC209, January, 2000)

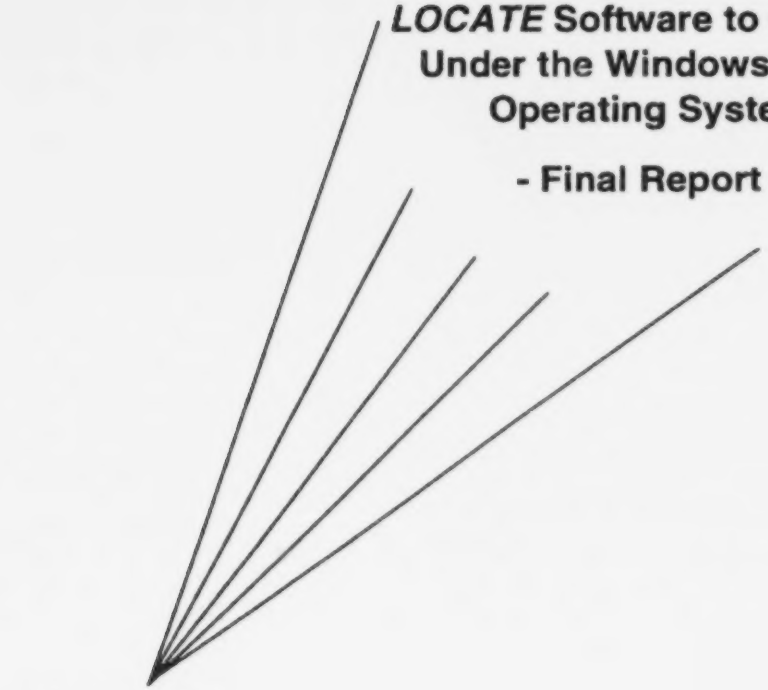
Contract: W7711-9-7572/001/TOR
"Application of Research to Port the *LOCATE* Software to Operate Under the Windows 97/NT Operating System"

Scientific Authority:
Mr. Keith C. Hendy
Simulation and Modelling for Acquisition, Rehearsal and Training (SMART)
Defence and Civil Institute of Environmental Medicine

© 2000 Her Majesty the Queen in right of Canada, as represented by the Minister of National Defence

Application of Research to Port the *LOCATE* Software to Operate Under the Windows 97/NT Operating System

- Final Report -



Artificial Intelligence
Management and Development Corporation
206 Keewatin Ave., Toronto, Ontario M4P 1Z8

Table of Contents

Background 1

Research Approach 2

Porting LOCATE from the Macintosh to the PC

Access to Functions Using the Right-Button of the PC Mouse 3

Screen Refresh Problem 5

Recommendations 7

References 8

APPENDICES

Appendix: Annotated List of LOCATE Source Code Files for the PC A-1

Background

LOCATE is a software tool for analyzing the effects of workspace layout on the quality of information exchange between humans and humans, and humans and machines (Hendy, 1984, 1989). LOCATE was developed in C code on the Macintosh computer using the platform independent graphic user interface tools provided by the Smart Elements® software environment. Although a Smart Elements development environment is said to be platform independent (PIGUI), the interface software must be ported to the machine and operating system on which it is to run. Unlike non-PIGUI developments, which may take up to two-thirds of the original development time to complete, this port is relatively straightforward.

The development has reached a point of maturity where DCIEM wishes to give a variety of users access to LOCATE. This is particularly so for users within DND design Directorates and researchers in the TTCP and NATO nations. To support this interchange of software, LOCATE must be ported to the PC environment that is prevalent in these domains.

Research Approach

The approach to this SOW involves the application of the Smart Elements tools to convert LOCATE to run under the Windows 97/NT operating system. Part of this port should ensure that right mouse button actions are functional in the LOCATE environment.

The report details the results of this port and the problems and limitations encountered, compares LOCATE's functionality on both the development and delivery platforms and makes recommendations for future ports.

Porting LOCATE from the Macintosh to the PC

The version of LOCATE ported to the PC contains all changes made to the development (Macintosh) version at the end of the contract entitled, "Testbed for Intelligent Aiding Using the LOCATE Workspace Layout Tool" (W7711-9-7546/001/TOR), completed at the end of October, 1999. Thus, all available intelligent help in the development version has been incorporated into the new PC version.

As anticipated, no problems were encountered in porting the system to the PC. The code transferred nicely from the Macintosh and all functions operable on the Macintosh are available on the PC.

Access to Functions Using the Right-Button of the PC Mouse

The fact that the PC supports a two-button mouse, while the Macintosh supports a single-button mouse is a difference not addressed in previous ports.

Hardware and software interface guidelines have not been as readily available for the PC as for the Macintosh and, in spite of recent attempts to provide those sources[1], very little was found to give clear guidance as to standard functions that might characterize the second mouse button of the PC.

Although no detailed information was found, a few common characteristics were inferred from a number of sources. Direct manipulation, e.g., clicking a mouse button, click-hold and dragging of a file, selecting (or highlighting) objects were seen as the function of the left-button mouse.

[1] See however, "The Windows Interface: An Application Design Guide," available from Microsoft Press.

A common function of the right-button mouse is to display a menu of options. Much less agreement exists on which particular functions to include, and decisions seem to depend on the problem domain for which a particular piece of software has been constructed. In general, the invoked menu allows the user to examine the properties of the object selected.

Edit functions, such as cut, copy, paste and duplicate, often are included. Rotation is another function that is seen in the menu. Other, secondary functions, such as selecting text, from the current text insertion position to the position of the cursor when a mouse button is clicked, occasionally appear but might equally well be assigned to the left button.

For the purpose of this port to LOCATE, the right button mouse invokes a menu with the following options for the context identified:

I. Click of Right Mouse Button in Free Space of Design

- Item Properties

- Paste

- Run Cost Function
- Optimize Workspace

II. Selected Elemental Workstation (EW) or Elemental or Fixed Obstruction (EOb; FOb)

- Item Properties

- Link Functions [for EW]
- Priority Weights [for EW]

- Minimize [boundary for EW, EOb, FOb]

III. Selected (any) Object

- Item Properties

- Rotate
- Ungroup (highlighted if a grouped object; dimmed if not)

- Cut
- Copy
- Paste

- Smart Help[2]

SCREEN REFRESH PROBLEM

This problem is characterized by delays in refreshing the screen when many links are present in a design. It is most obvious of course in displays of designs with a large number of objects, such as the Bridge Study designs. This problem, though prevalent on the PC, is not seen on the Macintosh.

In addition to slower performance with a large number of objects, the refresh process is accompanied by faint sounds as each link is redrawn. The source of those sounds has not been determined.

The problem can be eliminated if the code that tells the system to update the portion of the screen containing a given link is removed. Unfortunately, removing that portion of the code causes LOCATE to improperly update the screen and so, does not constitute a "fix" for the problem.

---

[2] This brings up the "Samrt Help" Window. Ultimately, this will be a context sensitive help feature.

The update area for each link is specified as a quadrilateral which includes the link itself and a couple of pixels on either side of the link. It is common in software to represent irregularly shaped regions as a collection of rectangles and, in the case of the link region in LOCATE, that can correspond to hundreds of single-pixel-wide rectangles for each link.

It is likely that the performance decrements in the PC are the result of the overhead required to track possibly thousands of rectangles. It is not yet clear whether this is due to some characteristic of the Smart Elements development environment or the PC Operating System, nor is it clear why the problem does not occur on the Macintosh.

The fix adopted in current port is to have LOCATE refresh the entire rectangle containing the link. Performance is enhanced, even though LOCATE must redraw a much larger portion of the screen. The trade-off is in the flicker created by refreshing this larger portion.

Recommendations

Three recommendations are offered for future ports to the PC. As performance has always been one of the strong points of LOCATE, additional effort should be expended to determine the source of the refresh problem. In spite of the substantially increased performance in this version, it is likely that further enhancements to system performance can be obtained if the true source(s) of the problem can be identified.

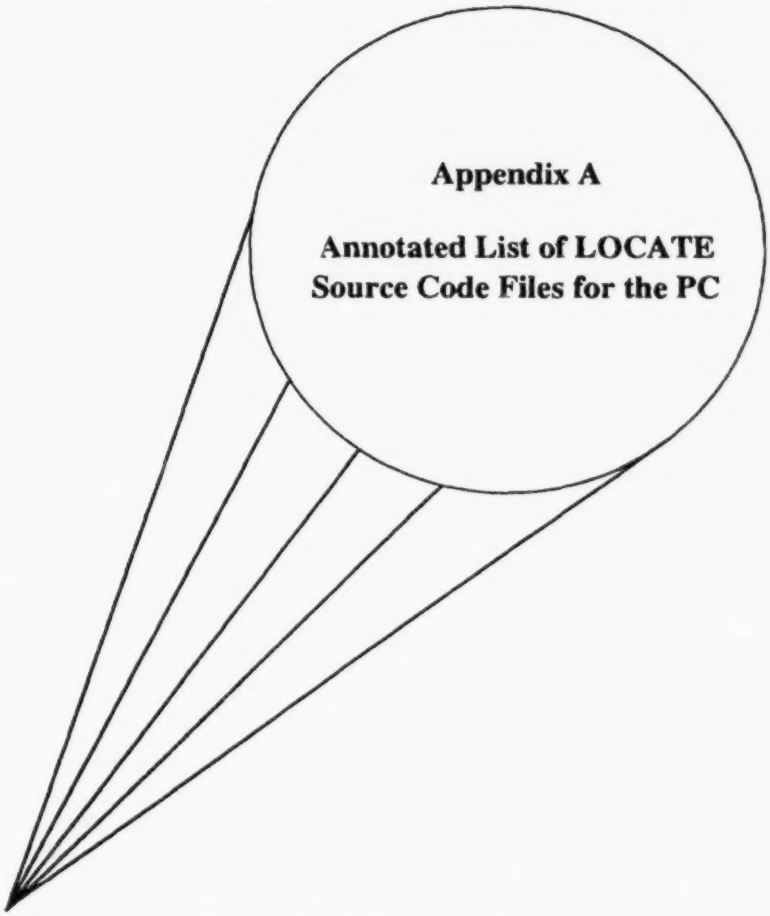
Second, as the Macintosh version is now being used as a testbed for incorporating both practical and experimental techniques for intelligent aiding, it may be useful to review the various elements to determine which functions should be part of a delivered system.

Lastly, DCIEM may want to consider restricting some of LOCATE's functionality in order to preserve the rights necessary for commercializing LOCATE in future. Such limits might include restricting the number of objects that users can create in a design and preventing users from saving their work. Such restrictions still allow users to develop an appreciation for the full range of LOCATE features.

References

Hendy, K. C. (1984). 'Locate': A program for computer-aided workspace layout. Master's Thesis, Department of Electrical Engineering, Monash University, Clayton, Victoria, Australia.

Hendy, K. C. (1989). A Model for Human-Machine-Human Interaction in Workspace Layout Problems. *Human Factors, 31(5),* 593-610.



Appendix A

Annotated List of LOCATE Source Code Files for the PC

Artificial Intelligence
Management and Development Corporation

Notes:
i) .c files are C source code
.obj files are compiled object code
.rc files are Open Interface resource description files
ii) Some extraneous files in the Locate folders are outdated and will be erased in future.

ALLOBJIN.C, ALLOBJIN.OBJ, ALLOBJIN.RC
- The "AllObjInfo" module contains the code and Open Interface (OI) resources necessary for the "All Objects Info" window.

AOBS.C, AOBS.OBJ, AOBS.RC
- The "AObs" module contains the code and OI resources necessary for the "Fixed Obstruction" window.

ASSIGN.C, ASSIGN.OBJ
- Original Locate C file

CFALERT.C, CFALERT.OBJ, CFALERT.RC
- The "CFAlert" module contains the code and OI resources necessary for the alert box that appears when there have been changes to the design but no cost function has been run since those changes have been made. The alert box appears before displaying the Cost Function History window .

CFBROWSE.C, CFBROWSE.OBJ, CFBROWSE.RC
- The "CFBrowse" module contains the code and OI resources necessary for the "Cost Function History" window.

CFCHECK.C, CFCHECK.OBJ, CFCHECK.RC
- The "CFCheck" module contains the code and OI resources necessary for the "Cost Function Checks" window.

COSTCOLR.C, COSTCOLR.OBJ, COSTCOLR.H, COSTCOLR.RC
- The "CostColr" module contains the code and Open Interface resources necessary for the "Cost Display Editor" window.

COSTDISP.C, COSTDISP.OBJ, COSTDISP.H, COSTDISP.RC
- The "CostDisp" module contains the code and Open Interface resources necessary for the "Cost Display" window.

COSTFN.C, COSTFN.OBJ, COSTFN.RC
- The "CostFn" module contains the code and OI resources necessary for the "Cost Function" window.

DRAWROTD.C, DRAWROTD.H, DRAWROTD.OBJ
- C code for handling the drawing of rotated objects

DXF.H
- Header file with DXF format constants

DXFOPT.C, DXFOPT.OBJ, DXFOPT.H
- The "DXFOpt" module contains the code and Open Interface resources necessary for the "DXF Import Options" window.

EDITOR.C, EDITOR.OBJ, EDITOR2.C, EDITOR2.OBJ, EDITOR.RC
- The "Editor" module contains the code and OI resources necessary for the main Locate window (includes code for Diagrammer, palette, rulers).

EVAL1.C, EVAL1.OBJ
- Original Locate C file

EWATTR.C, EWATTR.OBJ, EWATTR.RC
- The "EWAttr" module contains the code and OI resources necessary for the "Elemental Workstation Attributes" window.

EXTERN.H
- Original Locate header file

FORMAT.H
- Original Locate header file

FUNCT1.C, FUNCT1.OBJ
- Original Locate C file

GOALOBJ.CPP
- Contains code for defining and handling the C++ goal object

HEADER.DXF
- Contains information that gets added to all exported DXF files

HELPALRT.C, HELPALRT.OBJ, HELPALRT.RC
- The "HelpAlrt" module contains the code and Open Interface resources necessary for displaying the "Help Message" windows.

HELPMORE.C, HELPMORE.OBJ, HELPMORE.RC
- The "HelpMore" module contains the code and Open Interface resources necessary for displaying the "Help Message" windows containing the "More" button.

IMPRTDXF.C, IMPRTDXF.OBJ
- C code for handling the importing of a workspace from DXF format

INFOUPD.C, INFOUPD.OBJ, INFOUPD.H, INFOUPD.RC
- The "InfoUpd" module contains the code and Open Interface resources necessary for the "Instructions for Data Entry" window.

INFOWIN.C, INFOWIN.OBJ
- C code for handling the "Object Info" window

INTFC.H
- Header file with prototypes for interface functions

LFSUMM.C, LFSUMM.OBJ, LFSUMM.RC
- The "LFSumm" module contains the code and Open Interface resources necessary for the "Link Function Summary" window.

LINKDISP.C, LINKDISP.OBJ, LINKDISP.RC
- The "LinkDisp" module contains the code and OI resources necessary for the "Link Display" window.

LOCATE.C, LOCATE.OBJ
- Based on the original LOCATE.C file, this contains the code necessary for loading in a workspace and for computing the cost function.

LOCATE.DAT
- OI compiled resources that are used by the Locate application at run-time.

LOCATE.EXE
- The Locate application

LOCATE.H
- Original Locate header file

LOCATE.PDB
- Locate project for Microsoft Visual C++ version 4

LOCATE.RC
- OI resources in text format

LOCNEW.C, LOCNEW.OBJ
- C code for handling the creation of a new workspace

LOCNEWEW.C, LOCNEWEW.OBJ
- C code for handling the creation and deletion of workstations and obstructions

LOCSAVE.C, LOCSAVE.OBJ
- C code for handling the saving of a workspace

LOCSVDXF.C, LOCSVDXF.OBJ
- C code for handling the saving of a workspace in DXF format

MAIN.C, MAIN.OBJ, MAIN.RC
- The "Main" module contains the "main" function which starts up the application.

MISC.C, MISC.OBJ
- Original Locate C file

MISCRSRC.RC
- The "MiscRsrc" module contains OI resources needed by the application (primarily menu and icon resources).

MULTIOBJ.C, MULTIOBJ.OBJ, MULTIOBJ.RC
- The "MultiObj" module contains the code and Open Interface resources necessary for the "Multiple Object Creation" window.

NEWUSER.C, NEWUSER.OBJ, NEWUSER.H, NEWUSER.RC
- The "NewUser" module contains the code and Open Interface resources necessary for the "About You" window.

OPT.C
- Optimizer code for changing positions and angles

OPTIM.C, OPTIM.OBJ
- Original Locate C file

OPTOPT.C, OPTOPT.RC
- The "OptOpt" module contains the code and Open Interface resources necessary for the "Optimizer Options" window.

OPTSET.C, OPTSET.OBJ, OPTSET.RC
- The "OptSet" module contains the code and Open Interface resources necessary for the "Optimizer Settings" window.

OPTSTAT.C, OPTSTAT.OBJ, OPTSTAT.RC
- The "OptStat" module contains the code and Open Interface resources necessary for the "Optimizer Status" window.

OPTSWAP.C, OPTSWAP.OBJ
- Optimizer code for swapping workstations

ORIGIN.C, ORIGIN.OBJ
- Original Locate C file

OTHEROBJ.C, OTHEROBJ.OBJ, OTHEROBJ.RC

- The "OtherObj" module contains the code and OI resources necessary for the "Other Object" window.

OUTPUT.C, OUTPUT.OBJ

- Original Locate C file

PALEDIT.C, PALEDIT.OBJ, PALEDIT.RC

- The "PalEdit" module contains the code and OI resources necessary for the "Palette Editor" window.

PRINTPRE.C, PRINTPRE.OBJ, PRINTPRE.RC

- The "PrintPrev" module contains the code and OI resources necessary for the "Print Preview" window.

RULER.C, RULER.OBJ, RULER.RC

- The "Ruler" module contains the code and OI resources necessary for the "Ruler" window.

SMRTHELP.C, SMRTHELP.OBJ, SMRTHELP.RC

- The "SmrtHelp" module contains the code and Open Interface resources necessary for the "Smart Help" window.

SPLASH.C, SPLASH.OBJ, SPLASH.RC

- The "Splash" module contains the code and OI resources necessary for the startup screen.

SPLASH2.C, SPLASH2.OBJ, SPLASH2.H, SPLASH2.RC

- The "Splash2" module contains the code and Open Interface resources necessary for the "More on Locate" window.

START.C, START.OBJ, START.RC

- The "Start" module contains the code and OI resources necessary for the usability "Start" window.

STARTUP.C, STARTUP.OBJ, STARTUP.RC

- The "Startup" module contains the code and Open Interface resources necessary for the help reminder at startup (currently disabled).

STARTLOG.C, STARTLOG.OBJ, STARTLOG.RC

- The "StartLog" module contains the code and Open Interface resources necessary for the startup window that allows the user to enter a user name for the help system.

SYSMODL.C, SYSMODL.OBJ, SYSMODL.RC

- The "SysModl" module contains the code and Open Interface resources necessary for the "System Model" window.

TASKMODL.C, TASKMODL.OBJ, TASKMODL.RC

- The "TaskModl" module contains the code and Open Interface resources necessary for the "Task Model" window.

USERMODL.C, USERMODL.OBJ, USERMODL.RC

- The "UserModl" module contains the code and Open Interface resources necessary for the "User Model" window.

WEBBROWS.C, WEBBROWS.OBJ, WEBBROWS.H, WEBBROWS.RC

- The "WebBrows" module contains the code and Open Interface resources necessary for the "Web Browser" window.

WOBS.C, WOBS.OBJ, WOBS.RC

- The "WObs" module contains the code and OI resources necessary for the "Elemental Obstruction" window.

WSATTR.C, WSATTR.OBJ, WSA.RC

- The "WSAttr" module contains the code and OI resources necessary for the "Workspace Attributes" window.

UNCLASSIFIED

SECURITY CLASSIFICATION OF FORM
(Highest classification of Title, Abstract, Keywords)

DOCUMENT CONTROL DATA

(Security classification of title, body of abstract and indexing annotation must be entered when the overall document is classified)

1. ORIGINATOR (the name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g., Establishment sponsoring a contractor's report, or tasking agency, are entered in section 12.)
Artificial Intelligence Management and Development Corporation
206 Keewatin Avenue, Toronto, ON M4P 1Z8
CANADA

2. DOCUMENT SECURITY CLASSIFICATION (overall security classification of the document including special warning terms if applicable)
UNCLASSIFIED

3. DOCUMENT TITLE (the complete document title as indicated on the title page. Its classification should be indicated be the appropriate abbreviation (S,C,R or U) in parentheses after the title.)
Application of research to port the LOCATE software to operate under the Windows 97/NT operating system — Final Report (U)

4. DESCRIPTIVE NOTES (the category of the document, e.g., technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.)
Final Report

5. AUTHOR(S) (Last name, first name, middle initial. If military, show rank, e.g. Burns, Maj. Frank E.)
Edwards, Jack L.

6. DOCUMENT DATE (month and year of publication of document)	7.a. NO. OF PAGES (total containing information. Include Annexes, Appendices, etc.)	7.b. NO. OF REFS. (total cited in document)
January 2000	14	2

8.a. PROJECT OR GRANT NO. (if appropriate, the applicable research and development project or grant number under which the document was written. Please specify whether project or grant)	8.b. CONTRACT NO. (if appropriate, the applicable number under which the document was written)
Project 6kc12	PW&GS W7711-8-7572/001/TOR

9.a. ORIGINATOR'S DOCUMENT NUMBER (the official document number by which the document is identified by the originating activity. This number must be unique to this document.)	9.b. OTHER DOCUMENT NO.(S) (any other numbers which may be assigned this document either by the originator or by the sponsor.)
AC209	DCIEM No. CR 2000-071

10. DOCUMENT AVAILABILITY (any limitation on further dissemination of the document, other than those imposed by security classification)

X	Unlimited distribution
	Distribution limited to defence departments and defence contractors; further distribution only as approved
	Distribution limited to defence departments and Canadian defence contractors; further distribution only as approved
	Distribution limited to government departments and agencies; further distribution only as approved
	Distribution limited to defence departments; further distribution only as approved
	Other

11. ANNOUNCEMENT AVAILABILITY (any limitation to the bibliographic announcement of this document. This will normally correspond to the Document Availability (10.) However, where further distribution (beyond the audience specified in 10) is possible, a wider announcement audience may be selected.)

12. SPONSORING ACTIVITY (the name of the department project office or laboratory sponsoring the research and development. Include the address.)
Defence and Civil Institute of Environmental Medicine
P.O. Box 2000, 1133 Sheppard Ave West, Toronto, ON M3M 3B9 CANADA

DSIS DCD03
HFD 09/94

UNCLASSIFIED

SECURITY CLASSIFICATION OF FORM
(Highest classification of Title, Abstract, Keywords)

13. ABSTRACT (a brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), (R), or (U). It is not necessary to include here abstracts in both official languages unless the text is bilingual).

LOCATE is a software tool for analyzing the effects of workspace layout on the quality of information exchange between humans and humans, and humans and machines (Hendy, 1984, 1989). LOCATE was developed in C code on the Macintosh computer using the platform independent graphic user interface tools provided by the Smart Elements® software environment. Although a Smart Elements development environment is said to be platform independent (PIGUI), the interface software must be ported to the machine and operating system on which it is to run. Unlike non-PIGUI developments, which may take up to two-thirds of the original development time to complete, this port is relatively straightforward.

The development has reached a point of maturity where DCIEM wishes to give a variety of users access to LOCATE. This is particularly so for users within DND design Directorates and researchers in the TTCP and NATO nations. To support this interchange of software, LOCATE must be ported to the PC environment that is prevalent in these domains.

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible, keywords should be selected from a published thesaurus, e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)

HUMAN ENGINEERING TOOLS

HUMAN MODELLING

WORKSPACE LAYOUT

WORKSPACE DESIGN

FACILITY LAYOUT

COMPUTER-AIDED DESIGN

LOCATE